Understanding the Software Development Lifecycle (SDLC)

A Detailed, Business-Friendly & Practical Guide

Prepared by AHOM Technologies

Introduction to SDLC

- SDLC stands for Software Development Lifecycle.
- It is a structured process used to plan, design, develop, test, deploy, and maintain software.
- SDLC helps teams manage complex software projects in an organized way.
- It is widely used by startups, enterprises, and IT service providers.

Why SDLC is Important for Businesses

- Defines clear project scope, timeline, and budget.
- Reduces risks caused by unclear requirements.
- Improves coordination between business and technical teams.
- Ensures delivery of reliable and scalable software solutions.

Complete SDLC Flow

- SDLC follows a step-by-step approach from idea to support.
- Each phase has defined goals and deliverables.
- Output of one phase becomes the input for the next phase.
- This structured flow minimizes errors and rework.

Planning & Requirement Analysis

- Planning defines business goals, scope, timeline, and resources.
- Feasibility and risk assessment are done at this stage.
- Requirement analysis gathers business and user needs.
- Clear documentation helps avoid scope creep and confusion.

System & UI/UX Design

- System architecture and workflows are designed.
- User interface and user experience are planned.
- Database structure and security measures are defined.
- Technology stack is finalized during this stage.

Development Phase

- Developers write code based on approved designs.
- Features and modules are implemented step by step.
- Coding standards and best practices are followed.
- Version control helps manage code changes efficiently.

Testing & Deployment

- Testing identifies bugs, errors, and performance issues.
- Functional, security, and performance testing are performed.
- Deployment releases software to the live environment.
- Post-launch monitoring ensures smooth performance.

Maintenance & Support

- Bug fixes are provided based on user feedback.
- Performance improvements and feature updates are released.
- Security patches are applied regularly.
- Maintenance ensures long-term software reliability.

SDLC Models & Key Benefits

- Popular models include Waterfall, Agile, Iterative, and DevOps.
- Choosing the right model depends on project requirements.
- SDLC improves project control and software quality.
- It leads to predictable delivery and higher customer satisfaction.